Neural Computing and Applications (2022) 34:6241–6256
https://doi.org/10.1007/s00521-021-06794-x

ORIGINAL ARTICLE

Accelerated duality-aware correlation filters for visual tracking

Libin Xu[1] · Mingliang Gao[1] · Zheng Liu[2] · Qilei Li[3] · Gwanggil Jeon[4]

Received: 31 May 2021 / Accepted: 25 November 2021 / Published online: 14 January 2022

Abstract

Correlation filters (CF) based tracking methods have attracted considerable attentions for their competitive performance. However, the inherent issues of boundary effect and filter degradation, as well as the scale variation, degrade the tracking accuracy. In addition, the frame-by-frame updating strategy limits the tracking speed, especially in those deep features-based CF trackers. To address these issues, we propose a novel tracker, namely Accelerated Duality-aware Correlation Filters (ADCF), in this paper. In the proposed tracker, dual correlation filters, *i.e.*, translation filter and scale filter, are designed for target localization and scale estimation, respectively. A spatio-temporal regularization term is employed to suppress the boundary effect and filter degradation. Moreover, a model updating strategy named Sparse learning-based Average Peak-to-Correlation Energy (S-APCE) is proposed to accelerate the tracking speed. Finally, an Alternating Direction Method of Multipliers (ADMM) formulation is developed to optimize the ADCF efficiently. Extensive experimental results over six tracking benchmarks prove that the proposed tracker outperforms the state-of-the-art (SOTA) trackers in tracking accuracy and speed.

Keywords Visual tracking · Correlation filters · Spatio-temporal regularization · Online model update

1 Introduction

Visual tracking is a fundamental problem in computer vision, and it has been applied in many applications, *e.g.*, video retrieval [2], robotic perception [22] and human-machine interaction [13]. Despite significant progress in recent years, visual tracking remains challenging due to numerous complicating factors in real scenarios [29, 36], *e.g.*, illuminations, background clutter, and scale variation.

Generally, current visual tracking models are divided into two categories, namely generative models and discriminative models [36]. In generative models, the tracking task is implemented via searching the best-matched window, while discriminative models discriminate the target patch from the background by learning a discriminative classifier. Among the discriminative models, correlation filters (CF)-based trackers [3, 7, 17, 21] have drawn extensive attention. The advantage of the CF lies in a circulant matrix structure exploited, which can be calculated effectively by point-to-point operations and Fast Fourier Transform (FFT).

Bolme et al. [3] pioneered CF in visual tracking by learning a Minimum Output Sum of Squared Error (MOSSE) between multiple training image patches and the ideal correlation response template specified by the user. Galoogahi et al. [14] proposed an improved version of MOSSE named Multi-Channel Correlation Filters (MCCF), which utilizes features in multiple channels to boost the tracking performance. The part-based CF [34, 37] and scale-adaptive CF [6, 12] were proposed to handle the occlusion and size change. In addition, the features in CF were studied, and more discriminative features are proposed, *e.g.*, Color Names (CN) [7, 23], HOG [9, 15] and deep features [39, 44]. Moreover, model update strategies [25, 26] were proposed to improve the tracking accuracy and robustness in the multimedia environment.

✉ Mingliang Gao
mlgao@sdut.edu.cn

[1] School of Electrical and Electronic Engineering, Shandong University of Technology, Zibo 255000, China

[2] School of Engineering, University of British Columbia Okanagan, Kelowna V1V 1V7, Canada

[3] School of Electronic Engineering and Computer Science, Queen Mary University of London, London E1 4NS, UK

[4] Department of Embedded Systems Engineering, Incheon National University, Incheon 22012, South Korea

Although CF has achieved great success in visual tracking, it remains a challenge to gain satisfying performance in unconstrained scenarios due to the inherent issues, *i.e.*, boundary effect [9, 51] and filter degradation [5, 23, 51]. Meanwhile, the scale variation of the target has severe impacts on the tracking accuracy, and this problem is far from solved [6, 40, 47]. Moreover, although deep features have been adopted in CF to promote the tracking accuracy, as explored in [8, 10, 20, 43], the tracking speed of these trackers is reduced significantly.

To address the issues above, we propose an efficient yet effective tracker, namely Accelerated Duality-aware Correlation Filters (ADCF). The main contributions are summarized as follows.

1. A duality-aware CF model which consists of a translation filter and a scale filter is built to improve the tracking performance. The former translation filter ensembles deep and handcrafted features to localize the target accurately, and the latter scale filter exploits handcrafted features to estimate the scale efficiently. Meanwhile, a spatio-temporal regularization term that employs prior information is introduced to suppress the boundary effect and filter degradation.
2. An effective model updating strategy named Sparse learning-based Average Peak-to-Correlation Energy (S-APCE) is proposed to accelerate the tracking speed.
3. An ADMM formulation is developed to optimize the ADCF model efficiently, in which each subproblem is ensured a closed-form solution.
4. Extensive evaluations on six challenging tracking benchmarks are conducted, and experimental results demonstrate the competitive performance of the proposed tracker compared with more than 20 SOTA trackers.

The rest of this work is organized as follows. In Sect. 2, an overview of the relevant prior work is presented. In Sect.3, the ADCF model is proposed, and an ADMM formulation is developed to optimize this model. Meanwhile, the scale estimation and model updating strategy are presented. In Sect. 4, extensive evaluations of the proposed tracker with SOTA trackers are presented. Finally, this work is concluded in Sect. 5.

2 Related work

We introduce the related work in a problem-orientated manner, *i.e.,* boundary effect, filter degradation, and scale variation.

2.1 Boundary effect

CF utilizes FFT in the training and tracking process with the underlying periodic assumption. Although this assumption guarantees the strategy of dense sampling to construct the circulant matrix from a training sample patch, leading to unwanted boundary effect [9]. To suppress the unwanted boundary effect, Danelljan et al. [9] put forward the Spatially Regularized Discriminative Correlation Filters (SRDCF) method, in which a spatial regularization component was added for penalizing the model coefficient with a predefined weight constraint. Unlike SRDCF method in which negative examples are limited to circular shifted patches, Background-Aware Correlation Filters (BACF) [15] multiplied the filter with a binary matrix directly to generate real negative and positive training examples for tracker training. Furthermore, Context-Aware Correlation Filters (CACF) [32] model incorporated the global context information and takes advantage of more negative samples to alleviate the unwanted boundary effect.

2.2 Filter degradation

The filter updated by the linear interpolation cannot adjust to ubiquitous appearance changes, leading to filter degradation [23]. The filter degradation can be tackled from many aspects. In terms of training set management, Danelljan et al. [11] put forward Efficient Convolution Operators (ECO) for tracking model. ECO model formulated a compact generative model of the training sample distribution. In terms of temporal restriction, Li et al. [20] proposed Spatial-Temporal Regularized Correlation Filters (STRCF). In terms of overfitting alleviation, Sun et al. [33] adopted the concept of Region Of Interest (ROI)-based pooling and proposed an ROI Pooled Correlation Filters (RPCF) tracker with equality constraints. In terms of tracking confidence verification, Wang et al. proposed Large Margin object tracking with Circulant Feature maps (LMCF) [35] tracker using a multimodal target detection method. Among these strategies, temporal regularization has been proved to be an effective and efficient way [23].

2.3 Scale variation

The scale variation of the target has a serious impact on tracking accuracy [6]. To tackle this problem, some recent trackers adopted either multi-scale spatial pyramid [6, 28] or part-based multiple filters [27] to estimate the optimal scale. Moreover, several SOTA trackers [15, 20, 43, 44] attempted to introduce more complex scale models to further improve the tracking accuracy. However, the computational efficiency drops sharply due to multiple filtering operations performed in a single frame. Especially for the trackers with deep features [8, 20, 43], they are extremely time-consuming because the scale estimation requires multi-scale convolutional features.

3 Proposed method

In this section, we firstly review the formulation of STRCF [20], which is the foundation of the proposed model. Then, we propose the ADCF model and develop ADMM [4] formulation to optimize it efficiently. Finally, the target localization and scale estimation method, and the model update strategy are presented.

3.1 Revisit STRCF

The goal of spatial-temporal regularized correlation filter (STRCF) [20] is to learn a correlation filter $\mathbf{f} \in \mathbb{R}^{M \times N \times K}$ with $M \times N$ size and K channels at the tth frame, from the sample $\mathbf{x} \in \mathbb{R}^{M \times N \times K}$. The desired output $\mathbf{y} \in \mathbb{R}^{M \times N}$ is the Gaussian-shaped response, which includes a label for each location in the sample. The correlation filter can be trained by minimizing the following objective function,

$$\arg\min_{f} \frac{1}{2}\left\| y - \sum_{k=1}^{K} x_t^k * f_t^k F \right\|^2 + \frac{1}{2}\sum_{k=1}^{K}\left\| w_t \odot f_t^k \right\|_F^2 + \frac{\mu}{2}\sum_{k=1}^{K}\left\| f_t^k - f_{t-1}^k \right\|_F^2, \tag{1}$$

where $*$ and $\odot$ denote the circular convolution and point-wise multiplication, respectively. k is the channel index. The spatial weight $\mathbf{w}$ acts on the filter $\mathbf{f}_t$ to alleviate boundary effect. $\left\| \mathbf{f}_t^k - \mathbf{f}_{t-1}^k \right\|_F^2$ is the temporal regularization term to restrict abrupt changes of the filter by penalizing the difference between the current (tth frame) and previous ($t-1$th frame) filters. μ is the temporal regularization parameter.

In STRCF, however, both the target localization and scale estimation are performed on the same feature space. The computational cost of such a tracking strategy is extremely expensive when using deep features [20]. Moreover, the STRCF model is updated in a frame-by-frame manner, resulting in a low frame rate, which is not suitable for real-time scenarios.

3.2 Accelerated duality-aware correlation filter model

Based on STRCF, we propose an ADCF tracking model, as shown in Fig. 1. The ADCF tracker consists of a translation filter and a scale filter. The translation filter exploits ensembles of deep and handcrafted features for accurate target localization, while the scale filter exploits handcrafted features for efficient scale estimation. Meanwhile, a developed spatio-temporal regularization term that employs prior information is introduced to suppress the boundary effect and filter degradation. Moreover, an efficient and effective model update strategy named S-APCE is proposed to accelerate the tracking speed. The overall process of the ADCF tracker is divided into two portions, namely training and detection.

Training. Features are extracted by the pre-trained VGG-Net and HOG at the tth frame. The translation filter and scale filter are trained in a duality-aware manner, sharing portion features, spatio-temporal regularization and desired output. Both filters are optimized by the developed ADMM formulation.

Detection. The translation filter and scale filter from the training frame (tth frame), and the feature maps from the detection frame ($t+1$th frame) are combined to calculate the cross-correlation. In this stage, the translation filter is designed for accurate target localization, while the scale filter is used for fast scale estimation based on the maximum value of the response map. Finally, the S-APCE strategy is proposed to update the learning rate (η) of the appearance model.

The objective function of the ADCF can be formulated as follows,

$$\begin{aligned} \arg\min_{f} \frac{1}{2}\left\| y - \sum_{k=1}^{K} x_t^k * f_t^k \right\|_F^2 + \frac{\lambda_1}{2}\sum_{k=1}^{K}\left\| w_t \odot f_t^k \right\|_F^2 \\ + \frac{\lambda_2}{2}\left\| w_t - w_{t-1} \right\|_F^2 + \frac{\mu}{2}\sum_{k=1}^{K}\left\| f_t^k - f_{t-1}^k \right\|_F^2 \end{aligned} \tag{2}$$

Where the first term is the least square term. The second term introduces a spatial regularization term to alleviate the boundary effect. The third item introduces prior information of the spatial weight $\mathbf{w}$ to avoid its abrupt changes and degradation. The fourth term introduces a temporal regularization term to avoid filter degradation. λ_1 and λ_2 are the spatial regularization parameters, and μ is the temporal regularization parameter.

To optimize the ADCF model, we introduce an auxiliary variable $\widehat{\mathbf{g}} = \sqrt{T}\mathbf{F}\mathbf{f}$. Here, the symbol $\widehat{\cdot}$ denotes the discrete Fourier transform (DFT) of a signal, $T = M \times N$ is the feature length, and $\mathbf{F} \in \mathbb{C}^{T \times T}$ is an orthonormal matrix to map any T dimensional vector to the Fourier domain. Then, Eq. (2) can be transformed into the Fourier domain,

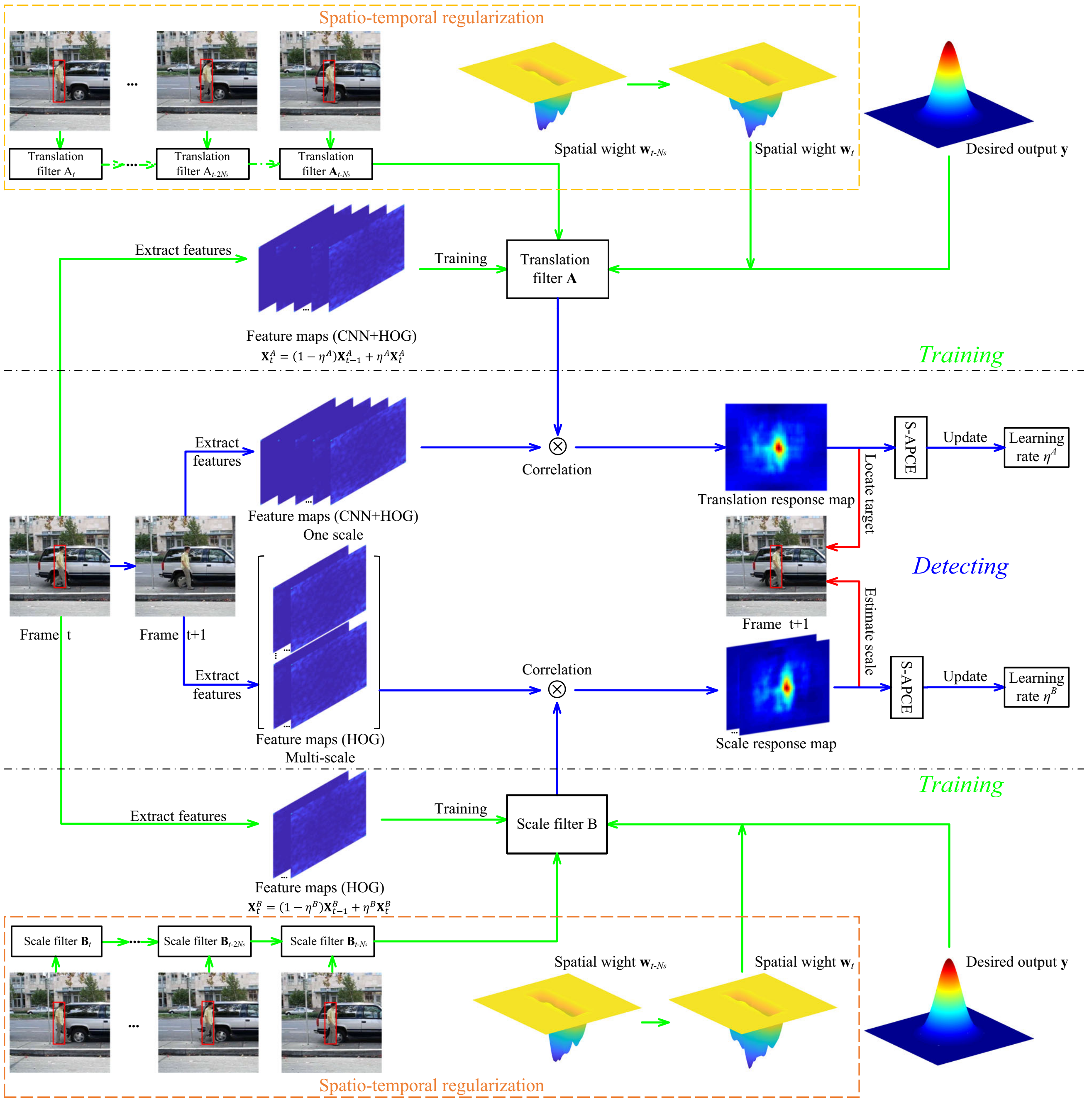

Spatio-temporal regularization
Translation filter A_t
Translation filter A_{t-2N_s}
Translation filter A_{t-N_s}
Spatial wight $\mathbf{w}_{t-Ns}$
Spatial wight $\mathbf{w}_t$
Desired output $\mathbf{y}$
Extract features
Training
Translation filter **A**
Feature maps (CNN+HOG)
$\mathbf{X}_t^A = (1-\eta^A)\mathbf{X}_{t-1}^A + \eta^A \mathbf{X}_t^A$
Training
Extract features
Feature maps (CNN+HOG) One scale
Correlation
Translation response map
S-APCE
Update
Learning rate η^A
Locate target
Detecting
Frame t
Frame t+1
Frame t+1
Estimate scale
Extract features
Feature maps (HOG) Multi-scale
Correlation
Scale response map
S-APCE
Update
Learning rate η^B
Training
Extract features
Training
Scale filter B
Feature maps (HOG)
$\mathbf{X}_t^B = (1-\eta^B)\mathbf{X}_{t-1}^B + \eta^B \mathbf{X}_t^B$
Scale filter $\mathbf{B}_t$
Scale filter $\mathbf{B}_{t-2N_s}$
Scale filter $\mathbf{B}_{t-N_s}$
Spatial wight $\mathbf{w}_{t-Ns}$
Spatial wight $\mathbf{w}_t$
Desired output $\mathbf{y}$
Spatio-temporal regularization

Fig. 1 Tracking framework of the proposed ADCF tracker

$$\begin{aligned}
&\underset{\mathbf{f}}{\operatorname{argmin}} \frac{1}{2}\left\| \widehat{\mathbf{y}} - \sum_{k=1}^{K} \widehat{\mathbf{x}}_t^k \odot \widehat{\mathbf{g}}_t^k \right\|_F^2 + \frac{\lambda_1}{2}\sum_{k=1}^{K}\left\|\mathbf{w}_t \odot \mathbf{f}_t^k\right\|_F^2 \\
&\quad + \frac{\lambda_2}{2}\left\|\mathbf{w}_t - \mathbf{w}_{t-1}\right\|_F^2 + \frac{\mu}{2}\sum_{k=1}^{K}\left\|\widehat{\mathbf{g}}_t^k - \widehat{\mathbf{g}}_{t-1}^k\right\|_F^2, \\
&\quad \text{s.t.,}\ \widehat{\mathbf{g}}_t^k = \sqrt{T}\mathbf{F}\mathbf{f}_t^k,\ k = 1,2,\ldots,K.
\end{aligned} \tag{3}$$

By minimizing Eq. (3), the optimal solution is obtained by

ADMM formulation [4]. The augmented Lagrangian form of Eq. (3) can be formulated as,

$$\begin{aligned}\mathcal{L} =& \frac{1}{2}\left\|\hat{\mathbf{y}} - \sum_{k=1}^{K}\hat{\mathbf{x}}_t^k \odot \hat{\mathbf{g}}_t^k\right\|_F^2 + \frac{\lambda_1}{2}\sum_{k=1}^{K}\|\mathbf{w}_t \odot \mathbf{f}_t^k\|_F^2 \\ &+ \frac{\lambda_2}{2}\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_F^2 \\ &+ \frac{\mu}{2}\sum_{k=1}^{K}\|\hat{\mathbf{g}}_t^k - \hat{\mathbf{g}}_{t-1}^k\|_F^2 + \frac{\gamma}{2}\sum_{k=1}^{K}\left\|\hat{\mathbf{g}}_t^k - \sqrt{T}\mathbf{F}\mathbf{f}_t^k\right\|_F^2 \\ &+ \sum_{k=1}^{K}\left(\hat{\mathbf{v}}_t^k\right)^{\mathrm{T}}\left(\hat{\mathbf{g}}_t^k - \sqrt{T}\mathbf{F}\mathbf{f}_t^k\right),\end{aligned} \tag{4}$$

where γ controls the step size for regularization, and $\mathbf{v}$ denotes the Lagrange multiplier. The superscript $\cdot^{\mathrm{T}}$ on a complex vector or matrix indicates conjugate transpose operation. By assigning $\mathbf{s} = \frac{1}{\gamma}\mathbf{v}$, the optimization of Eq. (4) can be reformulated as,

$$\begin{aligned}\mathcal{L} =& \frac{1}{2}\left\|\hat{\mathbf{y}} - \sum_{k=1}^{K}\hat{\mathbf{x}}_t^k \odot \hat{\mathbf{g}}_t^k\right\|_F^2 + \frac{\lambda_1}{2}\sum_{k=1}^{K}\|\mathbf{w}_t \odot \mathbf{f}_t^k\|_F^2 \\ &+ \frac{\lambda_2}{2}\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_F^2 \\ &+ \frac{\mu}{2}\sum_{k=1}^{K}\|\hat{\mathbf{g}}_t^k - \hat{\mathbf{g}}_{t-1}^k\|_F^2 \\ &+ \frac{\gamma}{2}\sum_{k=1}^{K}\left\|\hat{\mathbf{g}}_t^k - \sqrt{T}\mathbf{F}\mathbf{f}_t^k + \hat{\mathbf{s}}_t^k\right\|_F^2.\end{aligned} \tag{5}$$

Then, the ADMM formulation is adopted by solving the following subproblems alternately.

Subproblem $\hat{\mathbf{g}}$: If $\mathbf{f}$, $\mathbf{w}$ and $\hat{\mathbf{s}}$ are given, the optimal $\hat{\mathbf{g}}^*$ can be estimated by solving the optimization problem as,

$$\begin{aligned}\hat{\mathbf{g}}^* =& \underset{\hat{\mathbf{g}}}{\operatorname{argmin}}\left\{\frac{1}{2}\left\|\mathbf{y} - \sum_{k=1}^{K}\hat{\mathbf{x}}_t^k \odot \hat{\mathbf{g}}_t^k\right\|_F^2 + \frac{\mu}{2}\sum_{k=1}^{K}\|\hat{\mathbf{g}}_t^k - \hat{\mathbf{g}}_{t-1}^k\|_F^2\right. \\ &\left.+ \frac{\gamma}{2}\sum_{k=1}^{K}\left\|\hat{\mathbf{g}}_t^k - \sqrt{T}\mathbf{F}\mathbf{f}_t^k + \hat{\mathbf{s}}_t^k\right\|_F^2\right\}.\end{aligned} \tag{6}$$

However, it is hard to optimize Eq. (6) because of the high computational complexity. So, We consider processing all K channels of each pixel j to simplify formulation as,

$$\begin{aligned}\mathcal{V}_j^*(\hat{\mathbf{g}}) =& \underset{\hat{\mathbf{g}}}{\operatorname{argmin}}\left\{\frac{1}{2}\|\hat{\mathbf{y}}_j - \mathcal{V}_j(\hat{\mathbf{x}}_t)^{\mathrm{T}}\mathcal{V}_j(\hat{\mathbf{g}}_t)\|_F^2\right. \\ &+ \frac{\mu}{2}\|\mathcal{V}_j(\hat{\mathbf{g}}_t) - \mathcal{V}_j(\hat{\mathbf{g}}_{t-1})\|_F^2 \\ &\left.+ \frac{\gamma}{2}\left\|\mathcal{V}_j(\hat{\mathbf{g}}_t) - \mathcal{V}_j(\sqrt{T}\mathbf{F}\mathbf{f}_t) + \mathcal{V}_j(\hat{\mathbf{s}}_t)\right\|_F^2\right\},\end{aligned} \tag{7}$$

where $\mathcal{V}_j(\hat{\mathbf{x}}_t) \in \mathbb{C}^{K\times 1}$ denotes the vector containing values of $\hat{\mathbf{x}}_t$ on pixel j. The solution of Eq. (7) is obtained as,

$$\mathcal{V}^*(\hat{\mathbf{g}}) = \frac{1}{\mu+\gamma}\left[\mathbf{I} - \frac{\mathcal{V}_j(\hat{\mathbf{x}}_t)\mathcal{V}_j(\hat{\mathbf{x}}_t)^{\mathrm{T}}}{\mu+\gamma+\mathcal{V}_j(\hat{\mathbf{x}}_t)^{\mathrm{T}}\mathcal{V}_j(\hat{\mathbf{x}}_t)}\right]\mathbf{p}, \tag{8}$$

where $\mathbf{I}$ is an identity matrix,

$$\mathbf{p} = \mathcal{V}_j(\hat{\mathbf{x}}_t)\hat{\mathbf{y}}_j + \mu[\mathcal{V}_j(\hat{\mathbf{g}}_{t-1})] + \gamma\left[\mathcal{V}_j(\sqrt{T}\mathbf{F}\mathbf{f}_t) - \mathcal{V}_j(\hat{\mathbf{s}}_t)\right]. \tag{9}$$

The derivation of Eq. (8) adopts the Sherman Morrsion formulation [30],

$$\left(\mathbf{A} + \mathbf{u}\mathbf{v}^{\mathrm{T}}\right)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{v}^{\mathrm{T}}\mathbf{A}^{-1}}{1+\mathbf{v}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{u}}, \tag{10}$$

where $\mathbf{u}$ and $\mathbf{v}$ are two column vectors, and $\mathbf{u}\mathbf{v}^{\mathrm{T}}$ is a rank-one matrix.

Subproblem $\mathbf{f}$: If $\hat{\mathbf{g}}$, $\mathbf{w}$ and $\hat{\mathbf{s}}$ are given, the optimal $\mathbf{f}^*$ is determined as,

$$\begin{aligned}\mathbf{f}^* &= \underset{\mathbf{f}}{\operatorname{argmin}}\left\{\frac{\lambda_1}{2}\|\mathbf{w}_t \odot \mathbf{f}_t^k\|_F^2 + \frac{\gamma}{2}\left\|\hat{\mathbf{g}}_t^k - \sqrt{T}\mathbf{F}\mathbf{f}_t^k + \hat{\mathbf{s}}_t^k\right\|_F^2\right\} \\ &= \left[\lambda_1\mathbf{W}_t^{\mathrm{T}}\mathbf{W}_t + \gamma T\mathbf{I}\right]^{-1}\gamma T\left(\mathbf{g}_t^k + \mathbf{s}_t^k\right) \\ &= \frac{\gamma T\left(\mathbf{g}_t^k + \mathbf{s}_t^k\right)}{\lambda_1(\mathbf{w}_t \odot \mathbf{w}_t) + \gamma T},\end{aligned} \tag{11}$$

where $\mathbf{W}_t = \mathrm{diag}(\mathbf{w}_t) \in \mathbb{R}^{T\times T}$ denotes diagonal matrix. $\mathbf{g}_t^k$ and $\mathbf{s}_t^k$ can be obtained by inverse discrete Fourier transform (IDFT) (*i.e.*, $\mathbf{g}_t^k = \frac{1}{\sqrt{T}}\mathbf{F}^{\mathrm{T}}\hat{\mathbf{g}}_t^k$ and $\mathbf{s}_t^k = \frac{1}{\sqrt{T}}\mathbf{F}^{\mathrm{T}}\hat{\mathbf{s}}_t^k$).

Subproblem $\mathbf{w}$: Given $\mathbf{f}$, $\hat{\mathbf{g}}$ and $\hat{\mathbf{s}}$, the optimal $\mathbf{w}^*$ can be obtained as,

$$\begin{aligned}\mathbf{w}^* &= \underset{\mathbf{w}}{\operatorname{argmin}}\left\{\frac{\lambda_1}{2}\sum_{k=1}^{K}\|\mathbf{w} \odot \mathbf{f}_t^k\|_F^2 + \frac{\lambda_2}{2}\|\mathbf{w}_t - \mathbf{w}_{t-1}\|_F^2\right\} \\ &= \left[\lambda_1\sum_{k=1}^{K}\left(\mathbf{N}_t^k\right)^{\mathrm{T}}\mathbf{N}_t^k + \lambda_2\mathbf{I}\right]^{-1}\lambda_2\mathbf{w}_{t-1} \\ &= \frac{\lambda_2\mathbf{w}_{t-1}}{\lambda_1\sum_{k=1}^{K}\mathbf{f}_t^k \odot \mathbf{f}_t^k + \lambda_2\mathbf{I}},\end{aligned} \tag{12}$$

where $\mathbf{N}_t^k = \mathrm{diag}\left(\mathbf{f}_t^k\right) \in \mathbb{R}^{T\times T}$.

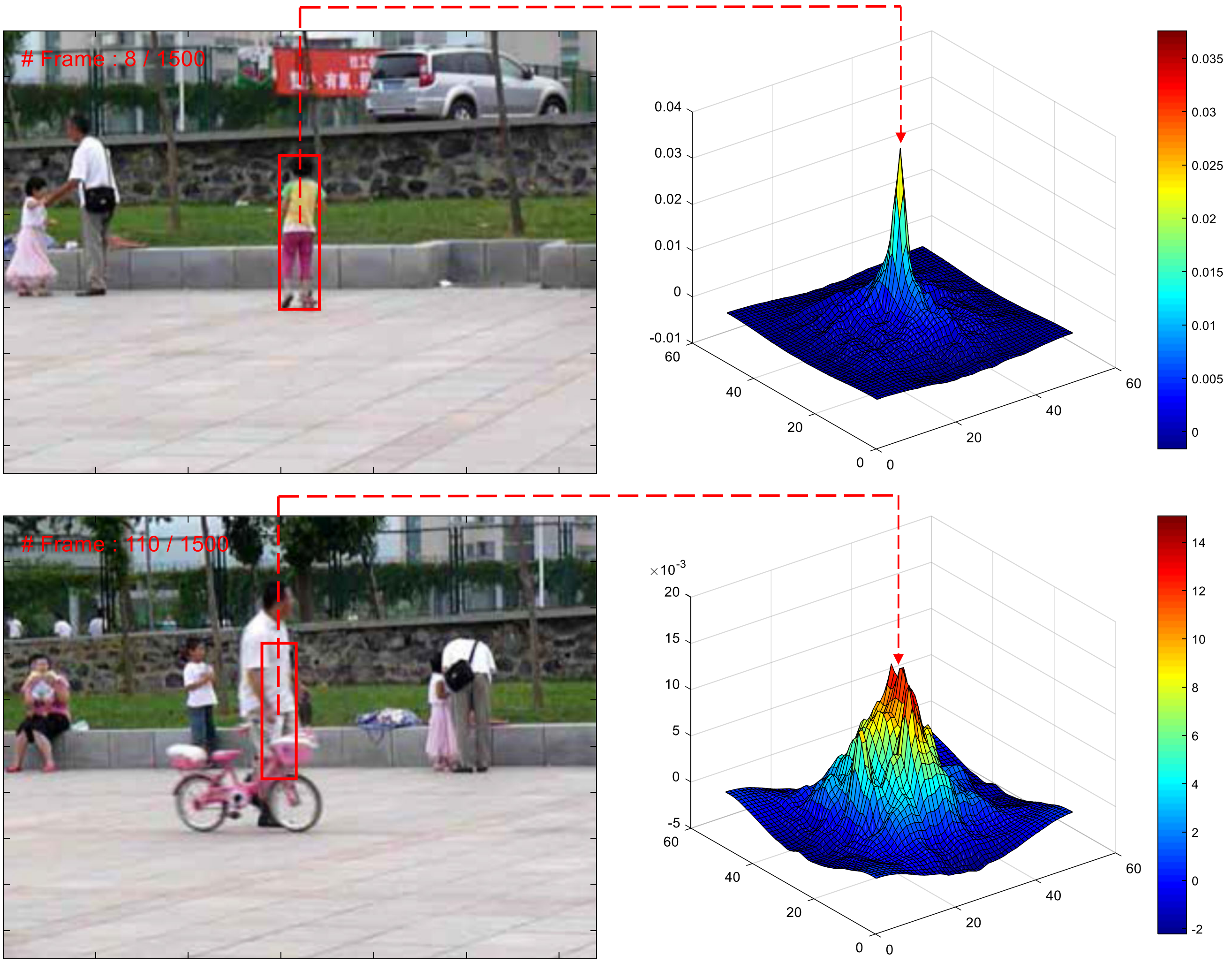


Fig. 2 The ideal response map (top), and non-ideal response map (bottom)

Lagrangian multiplier update: The Lagrange multiplier can be updated as,

$$\widehat{\mathbf{s}}^{i+1} = \widehat{\mathbf{s}}^{i} + \widehat{\mathbf{g}}^{i+1} - \widehat{\mathbf{f}}^{i+1}, \tag{13}$$

where i is the iteration index.

By solving the subproblems above iteratively, the objective function can be optimized effectively. Then the optimal filter $\widehat{\mathbf{g}}_t$ is utilized for detecting at the $t+1$th frame.

3.3 Target localization and scale estimation

The response map of the target in the Fourier domain is defined as,

$$\widehat{\mathbf{R}}_t = \sum_{k=1}^{K} \widehat{\mathbf{x}}^k \odot \widehat{\mathbf{g}}^k_{t-1}, \tag{14}$$

where $\mathbf{x}^k$ denotes candidate area, and $\mathbf{g}^k_{t-1}$ denotes trained filter from last frame.

ADCF is designed in a duality-aware manner in which the translation filter is designed for target localization, and the scale filter is for scale estimation. The translation filter is trained on ensembles of deep CNN features and handcrafted features (HOG feature in this work). Although the feature extraction is time-consuming, it merely executes on a single-scale search region during the tracking process. After obtaining the translation response map using Eq. (14), then, the target is localized by the maximum value of the response map.

Unlike the translation filter, which ensembles deep and handcrafted features, the scale filter exploits handcrafted features (HOG feature in this work) to estimate the scale efficiently. We apply the scale filter on five scale search regions and obtain their response maps using Eq. (14). Then, the best scale is the scale corresponding to the maximum value of the scale response maps. This proposed strategy can reduce the computational complexity efficiently under the premise of high tracking accuracy. This inference is verified in the ablation study in Sect. 4.5.

3.4 Model update

Most existing trackers update their models without considering the accuracy of detection [8, 9, 15, 20]. When the target is detected inaccurately, it will lead to a deterministic failure, *e.g.*, in the case of severely occluded.

Generally, the peak value and the fluctuation of the response map can reflect the confidence of the tracking results to a certain extent [35]. To this aim, we depict the ideal response map and the non-ideal response map of a target in Fig. 2. It depicts that when the detected target is completely matched with the correct target, the ideal response map should be unimodal, and other areas are smooth. Wang et al. [35] first proposed a high-confidence update strategy by adopting the maximum response value and an Average Peak-to-Correlation Energy (APCE) measure in LMCF tracker. The maximum response value is defined as,

$$\mathbf{R}_{max} = \max \mathcal{F}^{-1}\left(\widehat{\mathbf{R}}\right), \tag{15}$$

where $\mathcal{F}^{-1}$ denotes the inverse Fourier transform. The APCE measure is defined as,

$$APCE = \frac{|\mathbf{R}_{\max} - \mathbf{R}_{\min}|^2}{\operatorname{mean}\left[\sum_{w,h}\left(\mathbf{R}_{w,h} - \mathbf{R}_{\min}\right)^2\right]}, \tag{16}$$

where $\mathbf{R}_{\max}$, $\mathbf{R}_{\min}$ and $\mathbf{R}_{w,h}$ denote the maximum, minimum and the wth row hth column elements of response $\mathbf{R}$, respectively.

APCE reflects the smoothness of the response maps and the confidence level of the detected targets. The APCE value will drop significantly when the target encounters aberrance (*i.e.*, aberrant training samples) such as occlusion and background clutter [35]. According to the description of the high-confidence update strategy proposed by the LMCF, if the judgment condition "the maximum response value and APCE value both reach the threshold" is satisfied, the model will be updated. Assuming that multiple consecutive frames satisfy the judgment conditions, a continual update strategy will be activated, which results in a lower frame rate and robustness degradation due to overfitting the recent frames. On the contrary, if the model is not updated for a long time, which will lead to model degradation.

To this end, we propose a sparse update strategy based on APCE measure. This model update strategy not only inherits the advantages of APCE measure, but also considers the effectiveness of training samples and the efficiency of model update. Specifically, APCE value and maximum response value are used to control the learning rate of the appearance model to ensure the effectiveness of the current sample. The model is updated at the interval of N_s frames after initialization. Where, the appearance model $\mathbf{X}$ of the ADCF is updated as,

$$\begin{gathered} \mathbf{X}_t = (1-\eta)\mathbf{X}_{t-1} + \eta\mathbf{X}_t, \\ \eta = \begin{cases} \eta^* & \text{if}(APCE > \zeta APCE_{\text{hm}}) \cap (\mathbf{R}_{\max} > \zeta \mathbf{R}_{\text{hm}}) \\ 0 & \text{otherwise} \end{cases} \end{gathered} \tag{17}$$

where η and η^* are the learning rate and the learning rate when the training samples are high quality, respectively. $APCE_{hm}$ and $\mathbf{R}_{\text{hm}}$ denote the historical mean value of $APCE$ and $\mathbf{R}$, respectively. ζ is a threshold parameter. It is worth noting that although the model is updated in a sparse manner, the appearance model is constantly updated to adjust the appearance changes of the target. This model update strategy is verified in Section 4.5. The overall tracking algorithm of the ADCF tracker is summarized in Algorithm 1.

Algorithm 1: Overall tracking algorithm of the ADCF tracker.

```
Input: Initial the target state (i.e., position $p_1$ and scale $s_1$) by ground truth at
        the first frame.
Output: Target state at the $t$th frame.
Initialize the hyperparameters of the ADCF.
for $t = 1 : end$ do
    Training
    if $(t == 1 || \mathrm{mod}(t-1, N_s) == 0)$ then
        1. Extract CNN and HOG feature maps for translation filter training, and
           HOG feature maps for scale filter training, respectively.
        2. Optimize the translation filter and scale filter at the $t$th frame using
           Eq. (8), Eq. (11), Eq. (12) and Eq. (13) in $N$ iterations.
    end
    Detection
    1. Crop multi-scale search regions centered at $p_t$ with $S$ scales based on the
       bounding box at the $t$th frame.
    2. Extract VGG-Net and HOG feature maps with one scale for target
       localization, and HOG feature maps with $S$ scales for scale estimation,
       respectively.
    3. Compute translation response map with one scale, and scale response maps
       with $S$ scales using Eq. (14), respectively.
    4. Estimate the target bounding box with position $p_{t+1}$ as the center and $s_{t+1}$
       as the scale size, based on the maximum value of response maps.
end
```

4 Experimental results and discussion

In this section, we evaluate the ADCF tracker on six tracking benchmarks, *i.e.*, OTB2013 [41], OTB2015 [42], TC128 [24], UAV123 [31], UAV123@10 fps [31] and VOT2016 [19]. First, the implementation details and evaluation metrics are described. Then, quantitative and qualitative evaluations of the proposed tracker with the SOTA trackers are presented. Meanwhile, a more sophisticated analysis of the ADCF tracker is proved through ablation studies.

4.1 Experimental setup

The proposed tracker is implemented using the mixed programming of MATLAB2017a with the MatConvNet toolbox[1] on a PC with CPU (Intel i7 9700k) and GPU (NVIDIA GTX 1080Ti).The Parameters of ADCF are set as follows.

(1) For the translation CF, the spatial regularization parameters are set as $\lambda_1 = 1.2$ and $\lambda_2 = 0.001$. The temporal regularization parameter is set as $\mu = 0.01$. One-scale CNN (Conv4-3 from VGG-16) and HOG feature map are exploited for target localization.

(2) For the scale CF, the spatial regularization parameters are set as $\lambda_1 = 1.2$ and $\lambda_2 = 0.001$. The temporal regularization parameter is set as $\mu = 0.01$. Five-scale HOG feature maps are adopted for scale estimation.

(3) We set model update interval $N_s = 5$, the iteration of ADMM optimization $N = 3$, the threshold parameter $\zeta = 0.7$, and the learning rate $\eta^* = 0.02$. The step size parameter γ is initialized to 1 and updated by $\gamma^{i+1} = \min(\gamma_{\max}, \beta\gamma^i)(\beta = 10, \gamma_{\max} = 10,000)$ [15].

To make a fair comparison, the parameters of the ADCF model are fixed, and the compared trackers employ the public codes or results provided by the original publications.

4.2 Evaluation metrics

For OTB2013 [41], OTB2015 [42], TC128 [24], UAV123@10fps [31] and UAV123 [31] tracking benchmarks, success rate and precision are utilized under the rule of One Pass Evaluation (OPE) [41, 42]. The success rate denotes the percentage of frames in which the Intersection Over Union (IOU) exceeds a threshold (Note that the IOU is sometimes called overlap). Given the tracked bounding box r_t and the ground truth bounding box r_g, the IOU is defined as,

$$IOU = \frac{\left| r_t \bigcap r_g \right|}{\left| r_t \bigcup r_g \right|}, \tag{18}$$

where $\bigcap$ and $\bigcup$ represent the intersection and union of two regions, respectively. $|\cdot|$ denotes the number of pixels in the region. The precision denotes the percentage of frames where the Center Location Error (CLE) is under a threshold. Area Under Curve (AUC) in success rate and Distance Precision (DP, denoted by the percentage of frames whose

[1] https://www.vlfeat.org/matconvnet/.

CLE $\leq$ 20 pixels) in precision are adopted as the evaluation metrics to rank the success rate and precision of different trackers. In this work, accuracy evaluation on these benchmarks is based on the One Pass Evaluation (OPE) rule [42]. Moreover, the tracking speed is measured by Frames Per Second (FPS).

For VOT2016 [19] tracking benchmark, three primary evaluation metrics, namely Accuracy (A), Robustness (R) and Expected Average Overlap (EAO), are adopted to evaluate the tracking performance. A is calculated as the average overlap during successful tracking periods, and R is the total number of failures. The EAO metric combines the raw values of per-frame accuracy and the failures in a principled manner and has a clear practical interpretation. Meanwhile, it measures the expected no-reset overlap of a tracker running on a short-term sequence, and addresses the problem of increased variance and bias of the average overlap due to variable sequence lengths on practical datasets.

4.3 Quantitative evaluations

4.3.1 Evaluation on OTB2013 and OTB2015

OTB2013 [41] tracking benchmark contains 50 fully annotated sequences with substantial variations. OTB2015 tracking benchmark [42] is the extension of OTB2013, which contains 100 sequences. These two OTB tracking benchmarks are annotated with 11 attributes, *i.e.*, out-plane rotation (OPR), illumination variation (IV), out-of-view (OV), scale variation (SV), background clutter (BC), in-plane rotation (IPR), deformation (DEF), motion blur (MB), occlusion (OCC), low resolution (LR) and fast motion (FM).

We compare the ADCF tracker with the recent SOTA trackers, including ECO [11], DeepSTRCF [20], MCCT [37], STRCF [20], MCPF [48], LADCF-HC [44], CFWCR [16], ADNet [45], MCCT-H [37], BACF [15], ECO-HC [11], UDT [38], ARCF [18], ARCF-H [18], UDT+ [38], AutoTrack [23], STAPLE_CA [32], fDSST [12], RSST [49] and RaF [46]. The success and precision plots of the evaluated trackers on these two OTB tracking benchmarks are shown in Fig. 3. The comparative results indicate that the ADCF performs competitively against all the SOTA trackers. It achieves the best score in AUC and DP, respectively. Accuracy and speed comparisons of the top-5 trackers on OTB2013 and OTB2015 are shown in Table 1 and Table 2, respectively. The comparative results show that ADCF achieves the fastest speed on OTB2013 and OTB2015 among the GPU-based trackers, which benefits from the proposed model update strategy.

To analyze the performance of the trackers in handling different challenges, the attribute-based evaluations are performed. Some representative results are depicted in Fig. 4. For the sequences with the appearance variation of the target itself, *i.e.*, DEF, OPR and SV attributes, ADCF achieves 0.656, 0.673 and 0.661 AUC scores. For the sequences with environmental challenge scenarios, *i.e.*, BC, IV and OCC attributes, the target encounters partial or complete disappearance, which adversely affects the tracking accuracy. ADCF achieves 0.684, 0.701 and 0.670 AUC scores in these attributes, which surpass the second-best tracker by 3.5%, 2.5% and 0.4%, respectively. The improvement can be attributed to two factors. On the one hand, the temporal regularization in ADCF achieves a balance between the current filter $\mathbf{f}_t$ and the latest filter $\mathbf{f}_{t-1}$ to prevent filter degradation when the appearance of the target changes drastically. On the other hand, the spatial regularization establishes a balance between the current weight $\mathbf{w}_t$ and the latest weight $\mathbf{w}_{t-1}$. Thus, it enables the $\mathbf{w}_t$ close to the $\mathbf{w}_{t-1}$ to avoid abrupt changes and degradation. Particularly, ADCF achieves the best AUC score (0.661) in SV attribute. This can be attributed to the design of the scale filter, which avoid the loss of some detailed information in the feature description due to the pooling operation.

4.3.2 Evaluation on TC128

TC128 benchmark [24] consists of 128 challenging color sequences. We provide a comprehensive comparison of the proposed ADCF with SOTA trackers, including ECO [11], ASRCF [5], MCCT [37], LADCF-HC [44], MCCT-H [37], STRCF [20], ECO-HC [11], CFWCR [16], MCPF [48], UDT+ [38], ARCF [18], UDT [38], AutoTrack [23], STAPLE_CA [32], ARCF-H [18], SAMF_CA [32], BACF [15], RSST [49], fDSST [12], DCF_CA [32], RaF [46] and MOSSE_CA [32]. Fig. 5 shows the comparative results. It shows that ADCF achieves the best score both in AUC and DP. It is worth mentioning that the ASRCF tracker also designs a scale filter. Different from ASRCF which only uses spatial regularization for translation filter, the proposed ADCF introduces spatio-temporal regularization into both translation filter and scale filter. Meanwhile, the S-APCE update strategy in ADCF results in greater accuracy and faster speed.

We calculate the accuracy and speed of the top-5 trackers on TC128 benchmark, and the results are depicted in Table 3. Among these top-ranked trackers, the ADCF achieves the best accuracy and the fastest speed simultaneously.

4.3.3 Evaluation on UAV123 and UAV123@10fps

UAV123 tracking benchmark [31] contains 123 aerial sequences with more than 110K frames. This benchmark is

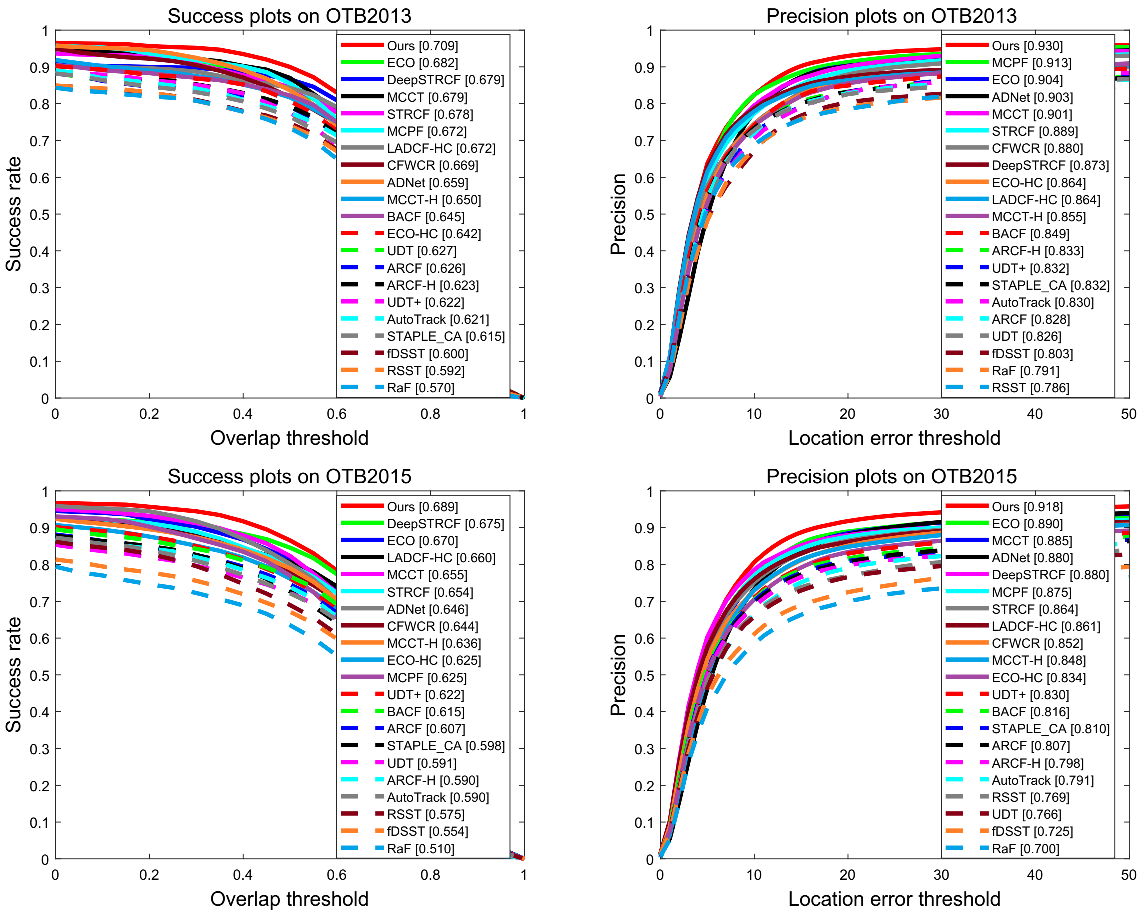


Fig. 3 Success and precision plots of the evaluated trackers on OTB2013 and OTB2015

Table 1 Comparative results of top-5 trackers on OTB2013 benchmark in accuracy and speed.

Trackers	DeepSTRCF	MCCT	STRCF	ECO	Ours
AUC	***0.679***	***0.679***	0.678	*0.682*	**0.709**
DP	0.873	***0.901***	0.889	*0.904*	**0.930**
FPS	***5.56***	2.60	*20.51*	1.85	**25.49**
CPU/GPU	GPU	GPU	CPU	GPU	GPU

The best three results are highlighted in bold, Italics and boldItalics, respectively

Table 2 Comparative results of top-5 trackers on OTB2015 benchmark in accuracy and speed.

Trackers	MCCT	DeepSTRCF	ECO	LADCF-HC	Ours
AUC	0.655	*0.675*	***0.670***	0.660	**0.689**
DP	***0.885***	0.880	*0.890*	0.861	**0.918**
FPS	2.60	***5.45***	1.82	*18.66*	**24.95**
CPU/GPU	GPU	GPU	GPU	CPU	GPU

The best three results are highlighted in bold, Italics and boldItalics respectively

more complex and challenging than other benchmarks as it includes significant changes in aspect ratio, abrupt camera motion, and severe occlusion. Meanwhile, UAV123@10fps [31] benchmark is a temporarily down-sampled version of the UAV123. Compared with the original UAV123 benchmark, UAV123@10fps is more challenging because the displacement of moving objects becomes bigger.

To further verify the performance of the ADCF tracker, comparative experiments are performed on UAV123 and UAV123@10fps benchmarks with SOTA trackers including ECO [11], DeepSTRCF [20], ASRCF [5], UDT+ [38], MCCT [37], ECO-HC [11], LADCF-HC [44], STRCF [20], UDT [38], AutoTrack [23], ARCF [18], RSST [49], BACF [15], MCCT-H [37], ARCF-H [18], STAPLE_CA [32], SAMF_CA [32], fDSST [12], DCF_CA [32], MOSSE_CA [32] and RaF [46]. The success and precision plots

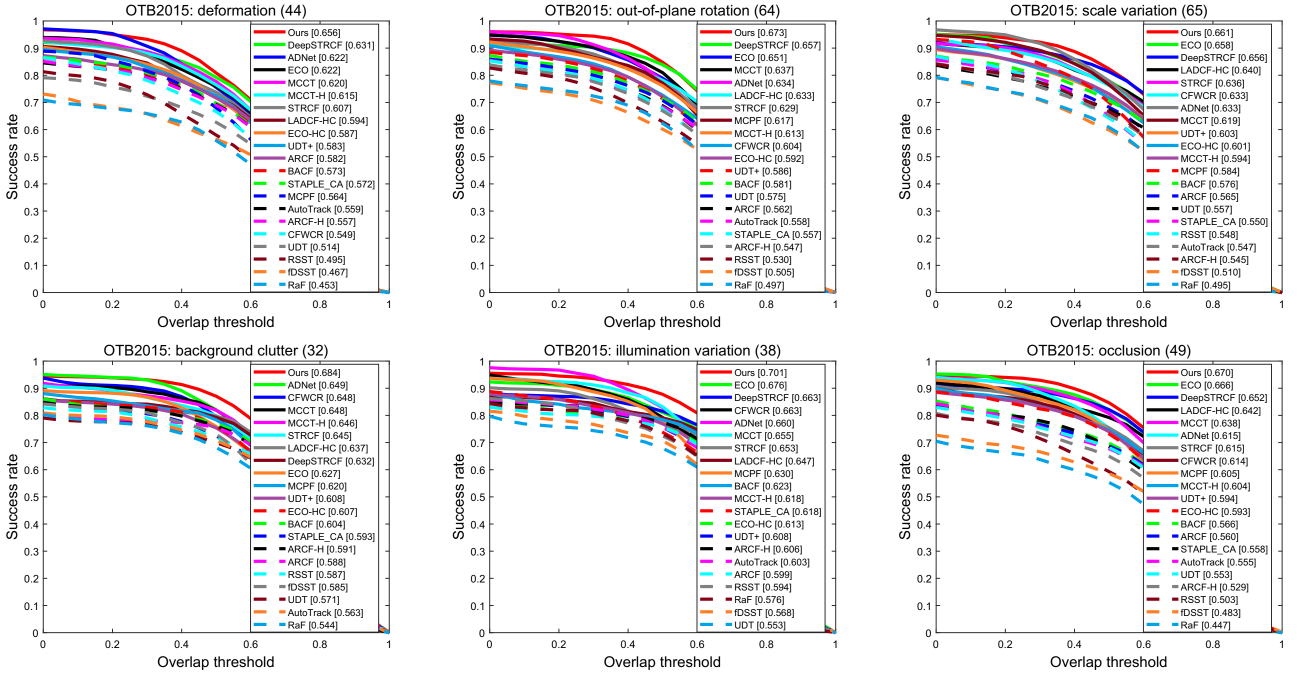


Fig. 4 Success plots of the evaluated trackers under 6 challenging attributes on OTB2015. The title of each sub-figure indicates the number of sequences marked with their attributes

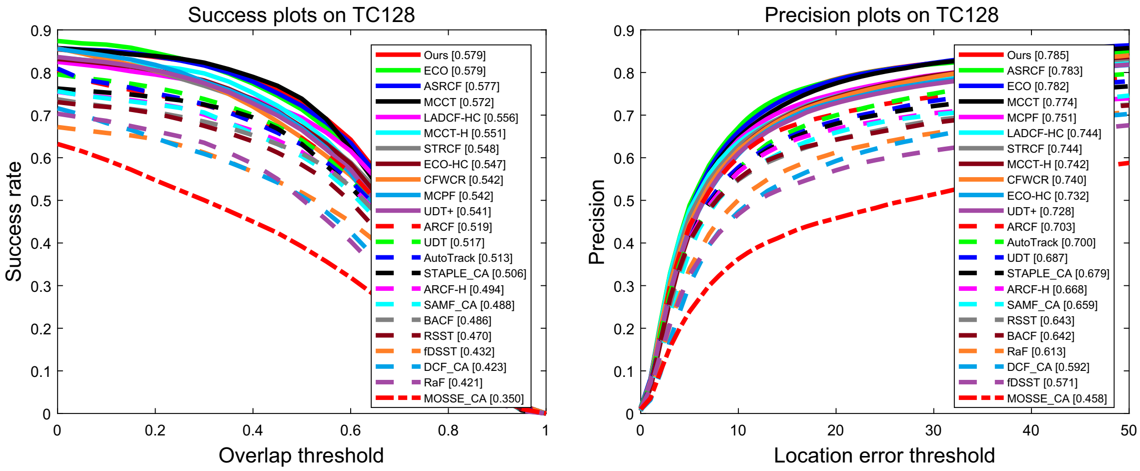


Fig. 5 Success and precision plots of the evaluated trackers on TC128

of these evaluated trackers are shown in Fig. 6. Meanwhile, the comparison of top-5 trackers in accuracy and speed are presented in Tables4 and 5, respectively. Generally, the AUC and DP scores of all the trackers on these two UAV benchmarks are worse than that on OTB [41, 42] and TC128 [24]. However, ADCF performs favorably against most trackers. Although the proposed tracker slightly underperforms ECO [11] and DeepSTRCF [20] in terms of accuracy, it is more than 12 times faster than ECO and 4 times faster than DeepSTRCF, respectively.

4.3.4 Evaluation on VOT2016

VOT2016 [19] tracking benchmark contains 60 challenging sequences. We compare the ADCF with top-10 trackers which are publicly listed in the VOT2016 official report [19], including C-COT [10], EBT [50], TCNN, Staple [1], STAPLE+, SSAT, MLDF, DDC, SRBT and DNT. The comparative results are shown in Table 6. As indicated in the VOT2016 report [19], the strict SOTA bound is 0.251 under EAO metrics, and the trackers whose EAO score exceeds this bound will be considered as SOTA

Table 3 Comparative results of top-5 trackers on TC128 benchmark in accuracy and speed

Trackers	MCCT	LADCF-HC	ASRCF	ECO	Ours
AUC	0.572	0.556	***0.577***	**0.579**	**0.579**
DP	0.774	0.744	*0.783*	***0.782***	**0.785**
FPS	2.65	***21.62***	*22.26*	1.82	**24.96**
CPU/GPU	GPU	CPU	GPU	GPU	GPU

The best three results are highlighted in bold, Italics and boldItalicse, respectively

trackers. Table 6 shows that the EAO score of the ADCF is 0.317 indicating that the ADCF is a SOTA tracker. Moreover, Table 6 shows that ADCF ranks second in accuracy and fourth in EAO and robustness, respectively.

Table 4 Comparative results of top-5 trackers on UAV123 benchmark in accuracy and speed

Trackers	DeepSTRCF	ASRCF	UDT+	ECO	Ours
AUC	***0.508***	***0.508***	0.502	**0.528**	*0.511*
DP	0.705	***0.738***	0.729	**0.749**	*0.743*
FPS	5.34	***20.45***	*20.94*	1.98	**25.31**
CPU/GPU	GPU	GPU	GPU	GPU	GPU

The best three results are highlighted in bold, Italics and boldItalics, respectively

4.4 Qualitative evaluations

The qualitative evaluations of the ADCF with nine SOTA trackers, *i.e.*, ARCF [18], ECO [11], BACF [15], AutoTrack [23], fDSST [12], UDT+ [38], MCCT [37], STRCF [20] and LADCF-HC [44] are shown in Fig. 7. Due to

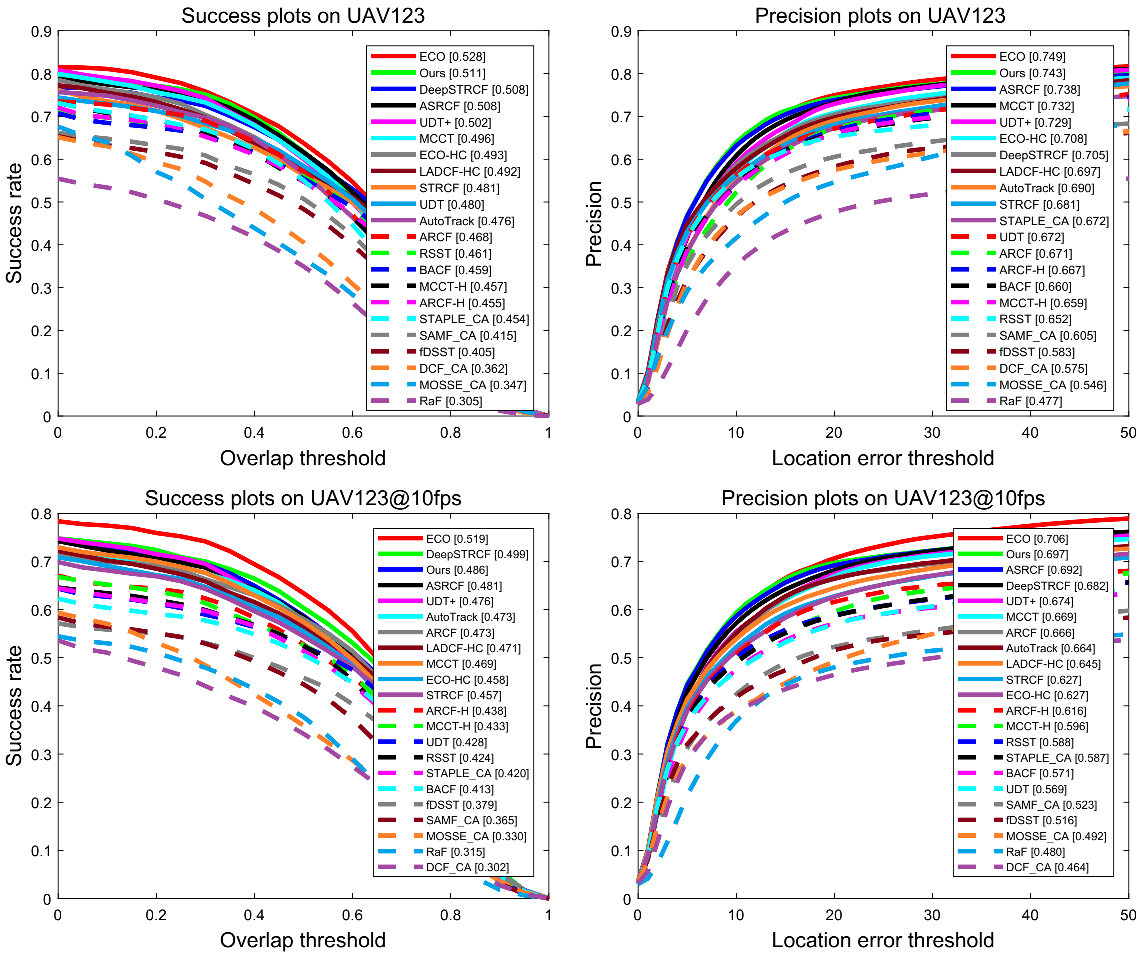


Fig. 6 Success and precision plots of the evaluated trackers on UAV123 and UAV123@10fps

Table 5 Comparative results of top-5 trackers on UAV123@10fps benchmark in accuracy and speed

Trackers	UDT+	DeepSTRCF	ASRCF	ECO	Ours
AUC	0.476	*0.499*	0.481	**0.519**	***0.486***
DP	0.674	0.682	***0.692***	**0.706**	*0.697*
FPS	***21.62***	5.41	*21.86*	1.91	**25.01**
CPU/GPU	GPU	GPU	GPU	GPU	GPU

The best three results are highlighted in bold, Italics and boldItalics, respectively

space constraints, we present representative results on ten challenging sequences from OTB2013 [41], OTB2015 [42], TC128 [24], UAV123 [31] and UAV123@10fps [31] tracking benchmarks. As shown in Fig. 7, the AutoTrack, ARCF and fDSST are less effective in handling scenarios with occlusion (Biker, Lemming and Ball_ce2) attributes. The LADCF-HC, STRCF, BACF and UDT+ trackers achieve poor performance on sequences with fast motion (Bird1, Skiing and uav1) attributes. Although MCCT and ECO perform well on most sequences, they are less effective when dealing with scale variation (Surf_ce1 and car18). The qualitative evaluation demonstrates that the proposed ADCF is more competitive than other trackers.

4.5 Ablation studies

First, we conduct an ablation study to demonstrate the effectiveness of the critical components in the ADCF tracker. The overall evaluation is shown in Table 7. The basic concepts are as follows. (1) “Baseline” refers to the method which does not adopt the spatio-temporal regularization, scale filter and S-APCE update strategy. (2) “Baseline+(STR)” denotes the baseline method by adding spatio-temporal regularization. (3) “Baseline+(SF)” denotes the baseline method by adding scale filter. (4) “Baseline+(S-APCE)” represents the baseline method by adopting the S-APCE update strategy. (5) “Baseline+(STR)+(SF)+(S-APCE)” is the final ADCF tracker.

As shown in Table 7, all the critical components, *i.e.*, scale filter, S-APCE update strategy and spatio-temporal regularization, contribute to the substantial improvement of the baseline method in terms of AUC and DP. Especially, the components of the scale filter and S-APCE update strategy boost the FPS by 38.4% and 83.7%. The final tracker improves the baseline method by 6.4%, 5.4% and 129.1% in AUC, DP and FPS, respectively.

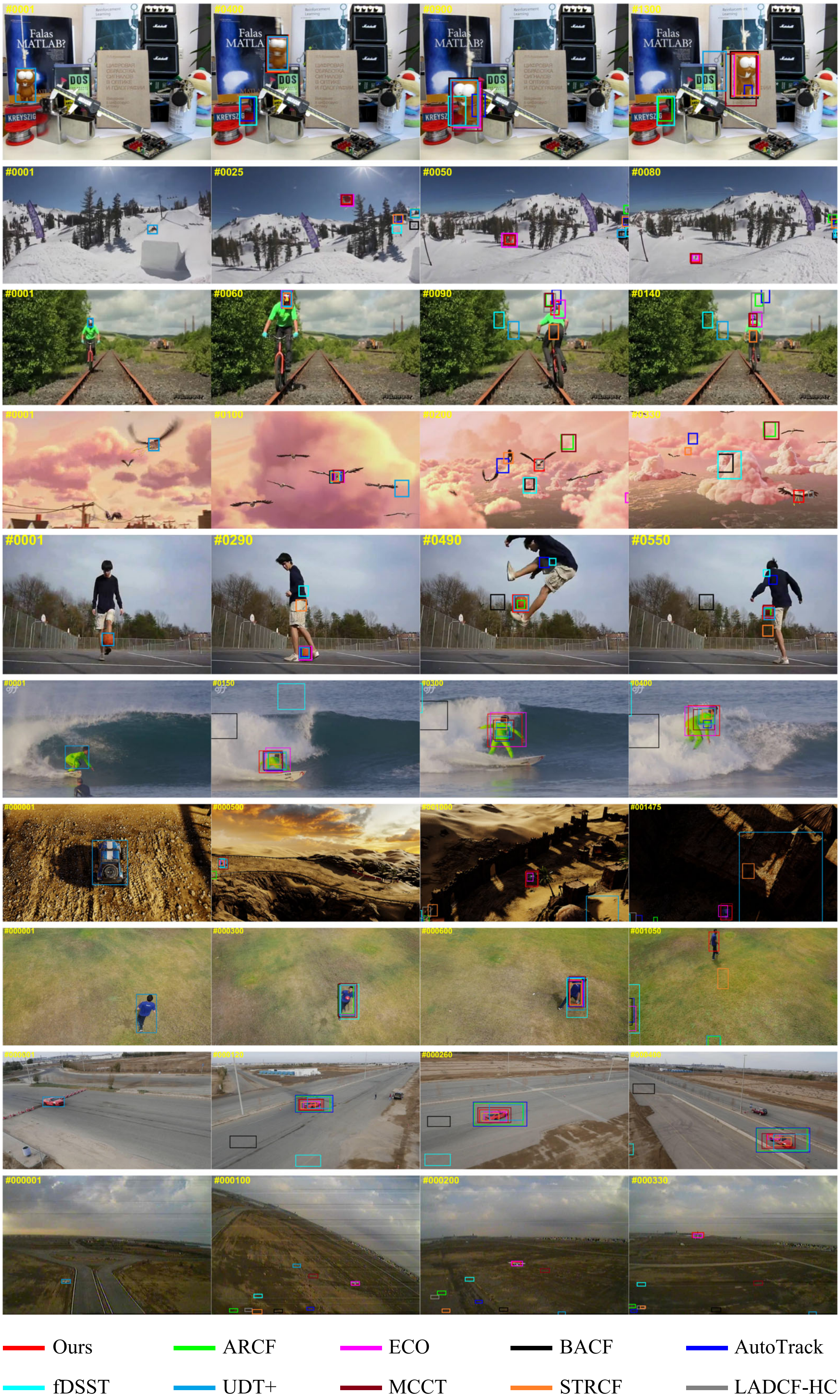
Second, we conduct an ablation study on the feature configurations in ADCF tracker. The tracking performance with different feature configurations are compared in terms of AUC and DP metrics on OTB2015 benchmark. As shown in Table 8, the fused features of HOG and Conv-4 outperform other feature configurations.

5 Conclusion

In this paper, we propose an accelerated duality-aware correlation filters (ADCF) model to improve the tracking performance. In the ADCF model, the translation filter exploits deep features to localize the target accurately, while the scale filter exploits handcrafted features to estimate the scale efficiently. Meanwhile, the spatial and temporal constraints are introduced into ADCF model to suppress the boundary effect and filter degradation simultaneously. Moreover, a model update strategy, namely sparse learning-based average peak-to-correlation energy (S-APCE), is proposed to update the ADCF model by the response map adaptively. Finally, an ADMM formulation is developed to optimize the ADCF model. Experiments are conducted on six challenging tracking benchmarks, *i.e.*, OTB2013, OTB2015, TC128, UAV123, UAV123@10fps and VOT2016. The qualitative and quantitative experimental results demonstrate the superiority of the proposed method against SOTA trackers in terms of tracking accuracy and speed.

Table 6 Comparison between ADCF and the top-10 trackers on VOT2016 benchmark in expected average overlap (EAO), accuracy (A) and robustness (R)

Trackers	C-COT	TCNN	SSAT	MLDF	Staple	DDC	EBT	SRBT	STAPLE+	DNT	Ours
EAO ↑	0.331	0.325	0.321	0.311	0.295	0.293	0.291	0.290	0.286	0.278	0.317
A ↑	0.539	0.554	0.577	0.490	0.544	0.541	0.465	0.496	0.557	0.515	0.570
R ↓	0.238	0.268	0.291	0.233	0.378	0.345	0.252	0.350	0.368	0.329	0.261


#0001
#0400
#0900
#1300
#0001
#0025
#0050
#0080
#0001
#0060
#0090
#0140
#0001
#0100
#0200
#0330
#0001
#0290
#0490
#0550
#0001
#0150
#0300
#0400
#000001
#000500
#001000
#001475
#000001
#000300
#000600
#001050
#000001
#000120
#000260
#000400
#000001
#000100
#000200
#000330
Ours
ARCF
ECO
BACF
AutoTrack
fDSST
UDT+
MCCT
STRCF
LADCF-HC

◀**Fig. 7** Qualitative evaluation of sample sequences from OTB2013, OTB2015, TC128, UAV123 and UAV123@10fps (from top to bottom: Lemming, Skiing, Biker, Bird1, Ball_ce2, Surf_ce1, car1_s, person8, car18 and uav1). The indices of the frames are marked on the top-left of each figure

Table 7 Ablation analysis of the key components in ADCF on OTB2015 benchmark

Trackers	AUC	DP	FPS
Baseline	0.647	0.871	10.89
Baseline+(STR)	0.673	0.892	7.40
Baseline+(SF)	0.665	0.887	15.07
Baseline+(S-APCE)	0.658	0.880	20.01
Baseline+(STR)+(SF)+(S-APCE)	**0.689**	**0.918**	**24.95**

Table 8 Tracking performance on OTB2015 with different feature configurations

	Features	AUC	DP
Hand-crafted	HOG	0.609	0.792
Hand-crafted+CNN	HOG+Conv-1	0.647	0.850
	HOG+Conv-2	0.650	0.859
	HOG+Conv-3	0.657	0.872
	HOG+Conv-4	**0.689**	**0.918**
	HOG+Conv-5	0.658	0.877

Acknowledgements Many thanks to the anonymous reviewers and editors for the valuable comments and suggestions. This work is partially supported by the National Natural Science Foundation of China (Nos. 61801272 and 61601266), and Natural Science Foundation of Shandong Province (Nos. ZR2020MF127 and ZR2021QD041).

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Bertinetto L, Valmadre J, Golodetz S, Miksik O, Torr PHS (2016) Staple: complementary learners for real-time tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1401–1409, 10.1109/CVPR.2016.156
2. Bhat G, Johnander J, Danelljan M, Khan FS, Felsberg M (2018) Unveiling the power of deep tracking. In: Proceedings of the european conference on computer vision, pp 493–509, https://doi.org/10.1007/978-3-030-01216-8_30
3. Bolme DS, Beveridge JR, Draper BA, Lui YM (2010) Visual object tracking using adaptive correlation filters. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2544–2550, https://doi.org/10.1109/CVPR.2010.5539960
4. Boyd S, Parikh N, Chu E, Peleato B, Eckstein J (2011) Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers, vol 3. Now Publishers Inc, MA
5. Dai K, Wang D, Lu H, Sun C, Li J (2019) Visual tracking via adaptive spatially-regularized correlation filters. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4665–4674, https://doi.org/10.1109/CVPR.2019.00480
6. Danelljan M, Häger G, Khan F, Felsberg M (2014) Accurate scale estimation for robust visual tracking. In: Proceedings of the British machine vision conference, https://doi.org/10.5244/C.28.65
7. Danelljan M, Khan FS, Felsberg M, v d Weijer J (2014) Adaptive color attributes for real-time visual tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1090–1097, https://doi.org/10.1109/CVPR.2014.143
8. Danelljan M, Häger G, Khan FS, Felsberg M (2015a) Convolutional features for correlation filter based visual tracking. In: Proceedings of the international conference on computer vision workshops, pp 621–629, https://doi.org/10.1109/ICCVW.2015.84
9. Danelljan M, Häger G, Khan FS, Felsberg M (2015b) Learning spatially regularized correlation filters for visual tracking. In: Proceedings of the international conference on computer vision, pp 4310–4318, https://doi.org/10.1109/ICCV.2015.490
10. Danelljan M, Robinson A, Shahbaz Khan F, Felsberg M (2016) Beyond correlation filters: Learning continuous convolution operators for visual tracking. In: Proceedings of the european conference on computer Vision, pp 472–488, https://doi.org/10.1007/978-3-319-46454-1_29
11. Danelljan M, Bhat G, Khan FS, Felsberg M (2017a) Eco: efficient convolution operators for tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 6931–6939, https://doi.org/10.1109/CVPR.2017.733
12. Danelljan M, Häger G, Khan FS, Felsberg M (2017) Discriminative scale space tracking. IEEE Trans Pattern Anal Mach Intell 39(8):1561–1575. https://doi.org/10.1109/TPAMI.2016.2609928
13. Fiaz M, Mahmood A, Javed S, Jung SK (2019) Handcrafted and deep trackers: Recent visual object tracking approaches and trends. ACM Comput Surv. https://doi.org/10.1145/3309665
14. Galoogahi HK, Sim T, Lucey S (2013) Multi-channel correlation filters. In: Proceedings of the international conference on computer vision, pp 3072–3079, https://doi.org/10.1109/ICCV.2013.381
15. Galoogahi HK, Fagg A, Lucey S (2017) Learning background-aware correlation filters for visual tracking. In: Proceedings of the international conference on computer vision, pp 1144–1152, https://doi.org/10.1109/ICCV.2017.129
16. He Z, Fan Y, Zhuang J, Dong Y, Bai H (2017) Correlation filters with weighted convolution responses. In: Proceedings of the international conference on computer vision workshops, pp 1992–2000, https://doi.org/10.1109/ICCVW.2017.233
17. Henriques JF, Caseiro R, Martins P, Batista J (2012) Exploiting the circulant structure of tracking-by-detection with kernels. In: Proceedings of the european conference on computer vision, pp 702–715, https://doi.org/10.1007/978-3-642-33765-9_50
18. Huang Z, Fu C, Li Y, Lin F, Lu P (2019) Learning aberrance repressed correlation filters for real-time uav tracking. In: Proceedings of the international conference on computer vision, pp 2891–2900, https://doi.org/10.1109/ICCV.2019.00298
19. Kristan M, Leonardis A, Matas Jea (2016) The visual object tracking vot2016 challenge results. In: Proceedings of the european conference on computer vision workshops, pp 777–823, https://doi.org/10.1007/978-3-319-46448-0_27
20. Li F, Tian C, Zuo W, Zhang L, Yang M (2018) Learning spatial-temporal regularized correlation filters for visual tracking. In:

Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4904–4913, 10.1109/CVPR.2018.00515

21. Li G, Peng M, Nai K, Li Z, Li K (2020) Multi-view correlation tracking with adaptive memory-improved update model. Neural Comput Appl 32(13):9047–9063. https://doi.org/10.1007/s00521-019-04413-4
22. Li P, Wang D, Wang L, Lu H (2018) Deep visual tracking: review and experimental comparison. Pattern Recogn 76:323–338. https://doi.org/10.1016/j.patcog.2017.11.007
23. Li Y, Fu C, Ding F, Huang Z, Lu G (2020) Autotrack: Towards high-performance visual tracking for uav with automatic spatio-temporal regularization. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 11920–11929, 10.1109/CVPR42600.2020.01194
24. Liang P, Blasch E, Ling H (2015) Encoding color information for visual tracking: algorithms and benchmark. IEEE Trans Image Process 24(12):5630–5644. https://doi.org/10.1109/TIP.2015.2482905
25. Liu S, Liu D, Muhammad K, Ding W (2021) Effective template update mechanism in visual tracking with background clutter. Neurocomputing 458:615–625. https://doi.org/10.1016/j.neucom.2019.12.143
26. Liu S, Wang S, Liu X, Gandomi AH, Daneshmand M, Muhammad K, De Albuquerque VHC (2021) Human memory update strategy: a multi-layer template update mechanism for remote visual monitoring. IEEE Trans Multimed 23:2188–2198. https://doi.org/10.1109/TMM.2021.3065580
27. Liu T, Wang G, Yang Q (2015) Real-time part-based visual tracking via adaptive correlation filters. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4902–4912, https://doi.org/10.1109/CVPR.2015.7299124
28. Ma C, Yang X, Chongyang Zhang, Yang M (2015) Long-term correlation tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5388–5396, https://doi.org/10.1109/CVPR.2015.7299177
29. Marvasti-Zadeh SM, Cheng L, Ghanei-Yakhdan H, Kasaei S (2021) Deep learning for visual tracking: a comprehensive survey. IEEE Trans Intell Transport Sys. https://doi.org/10.1109/TITS.2020.3046478
30. Morrison SWJ (1950) Adjustment of an inverse matrix corresponding to a change in one element of a given matrix. Ann Maths Stat 21(1):124–127. https://doi.org/10.2307/2236561
31. Mueller M, Smith N, Ghanem B (2016) A benchmark and simulator for uav tracking. In: Proceedings of the european conference on computer vision, pp 445–461, https://doi.org/10.1007/978-3-319-46448-0_27
32. Mueller M, Smith N, Ghanem B (2017) Context-aware correlation filter tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1387–1395, https://doi.org/10.1109/CVPR.2017.152
33. Sun Y, Sun C, Wang D, He Y, Lu H (2019) Roi pooled correlation filters for visual tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5776–5784, https://doi.org/10.1109/CVPR.2019.00593
34. Tang M, Yu B, Zhang F, Wang J (2018) High-speed tracking with multi-kernel correlation filters. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4874–4883, https://doi.org/10.1109/CVPR.2018.00512
35. Wang M, Liu Y, Huang Z (2017) Large margin object tracking with circulant feature maps. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4800–4808, https://doi.org/10.1109/CVPR.2017.510
36. Wang N, Shi J, Yeung D, Jia J (2015) Understanding and diagnosing visual tracking systems. In: Proceedings of the IEEE international conference on computer vision, pp 3101–3109, https://doi.org/10.1109/ICCV.2015.355
37. Wang N, Zhou W, Tian Q, Hong R, Wang M, Li H (2018) Multi-cue correlation filters for robust visual tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4844–4853, https://doi.org/10.1109/CVPR.2018.00509
38. Wang N, Song Y, Ma C, Zhou W, Liu W, Li H (2019a) Unsupervised deep tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1308–1317, https://doi.org/10.1109/CVPR.2019.00140
39. Wang P, Sun M, Wang H, Li X, Yang Y (2020) Convolution operators for visual tracking based on spatial-temporal regularization. Neural Comput Appl 32(10):5339–5351. https://doi.org/10.1007/s00521-020-04704-1
40. Wang X, Hou Z, Yu W, Jin Z, Zha Y, Qin X (2019) Online scale adaptive visual tracking based on multilayer convolutional features. IEEE Trans Cybernet 49(1):146–158. https://doi.org/10.1109/TCYB.2017.2768570
41. Wu Y, Lim J, Yang M (2013) Online object tracking: A benchmark. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2411–2418, https://doi.org/10.1109/CVPR.2013.312
42. Wu Y, Lim J, Yang M (2015) Object tracking benchmark. IEEE Trans Patt Anal Mach Intell 37(9):1834–1848. https://doi.org/10.1109/TPAMI.2014.2388226
43. Xu T, Feng Z, Wu X, Kittler J (2019a) Joint group feature selection and discriminative filter learning for robust visual object tracking. In: Proceedings of the international conference on computer vision, pp 7949–7959, https://doi.org/10.1109/ICCV.2019.00804
44. Xu T, Feng Z, Wu X, Kittler J (2019) Learning adaptive discriminative correlation filters via temporal consistency preserving spatial feature selection for robust visual object tracking. IEEE Trans Image Process 28(11):5596–5609. https://doi.org/10.1109/TIP.2019.2919201
45. Yun S, Choi J, Yoo Y, Yun K, Choi JY (2017) Action-decision networks for visual tracking with deep reinforcement learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1349–1358, https://doi.org/10.1109/CVPR.2017.148
46. Zhang L, Varadarajan J, Suganthan PN, Ahuja N, Moulin P (2017a) Robust visual tracking using oblique random forests. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5825–5834, https://doi.org/10.1109/CVPR.2017.617
47. Zhang S, Lu W, Xing W, Zhang L (2020) Learning scale-adaptive tight correlation filter for object tracking. IEEE Trans Cybernet 50(1):270–283. https://doi.org/10.1109/TCYB.2018.2868782
48. Zhang T, Xu C, Yang M (2017b) Multi-task correlation particle filter for robust object tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4819–4827, https://doi.org/10.1109/CVPR.2017.512
49. Zhang T, Xu C, Yang M (2019) Robust structural sparse tracking. IEEE Trans Patt Anal Machine Intell 41(2):473–486. https://doi.org/10.1109/TPAMI.2018.2797082
50. Zhu G, Porikli F, Li H (2016) Beyond local search: tracking objects everywhere with instance-specific proposals. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 943–951, https://doi.org/10.1109/CVPR.2016.108
51. Zhu G, Zhang Z, Wang J, Wu Y, Lu H (2019) Dynamic collaborative tracking. IEEE Trans Neural Networks Learn Sys 30(10):3035–3046. https://doi.org/10.1109/TNNLS.2018.2861838

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.